Course : CSE 3200

Topic: System Development Project

Face Mask Detection

SUPERVISED BY

Md. Abdus Salim Mollah
Assistant Professor,
Department of Computer Science & Engineering
Khulna University of Engineering & Technology (KUET)
Khulna -9203, Bangladesh.

PRESENTED BY

Shafin Junayed(Roll-1707115)
Muhammad Faiaz Rahman Nafee(Roll-1707077)

Presentation Outline

1.introduction

2.Objectives

3.Implementation

4.Main Architecture

5.Software Needed and Language used

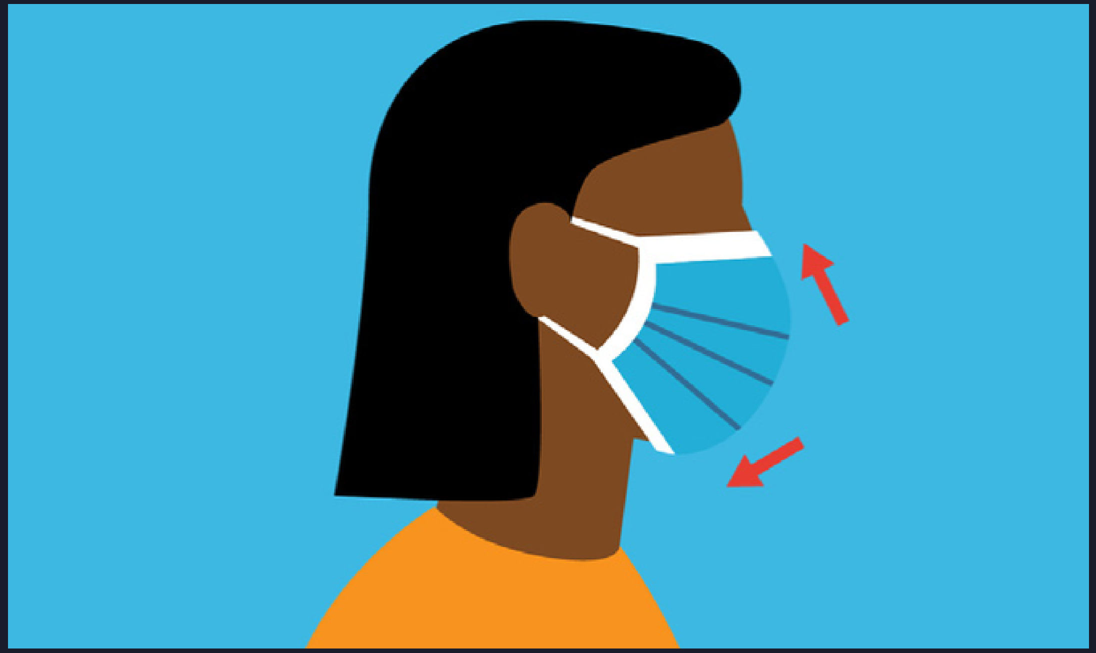
6.Feature

7.Dataset and performance Analysis

8.Future works

9.Conclusion

Introduction



The rapid outbreak of COVID-19 has caused serious harm and infected tens of millions of people worldwide. Since there is no specific treatment, wearing masks has become an effective method to prevent the transmission of COVID-19 and is required in most public areas, which has also led to a growing demand for automatic real-time mask detection services to replace manual reminding. However, few studies on face mask detection are being conducted. It is urgent to improve the performance of mask detectors. In this system, we made the mask detector to identify the person having the mask on or not.

Objectives

1.Familiarize with the OpenCV,MobileNet v2,Numpy,Tensorflow,Matplotlib,Cuda, and Keras.

2.Gain knowledge about how CNN works.

3.Able to make model with labels and features.

4.Determine the graph overfitting or underfitting.

5.Able to gain knowledge about the different function of deep learning.

6.Gain knowledge about how to train data.

7.Familiarize with different kind of layers.

Implementation

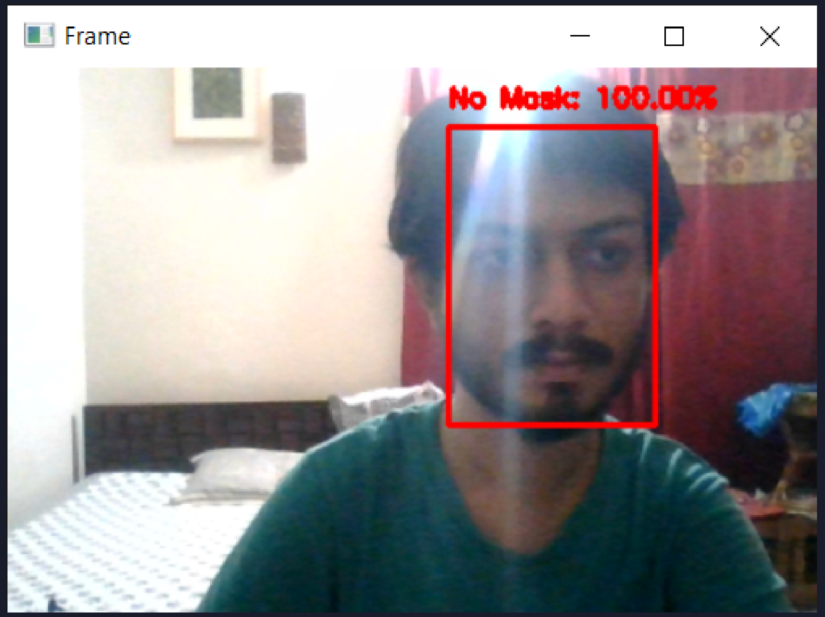
First of all, all we need to install the tensorflow, keras, OpenCV,Numpy,Cuda and Matplotlib.The any cmd or command form, we need to install the anaconda to run the code.Here, we use the pip function to proceed. Then, there are 3 python code named

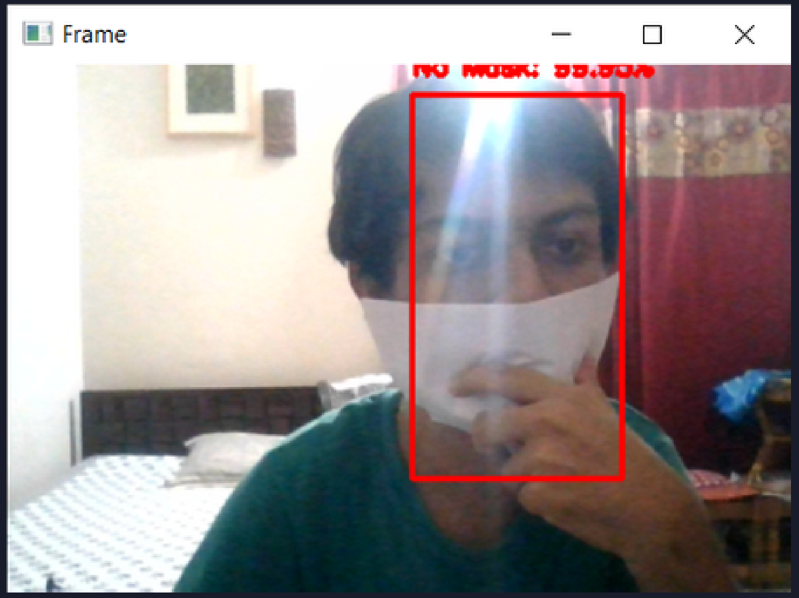
- detect_mask_image.py
- detect_mask_video.py
- train_mask_detector.py

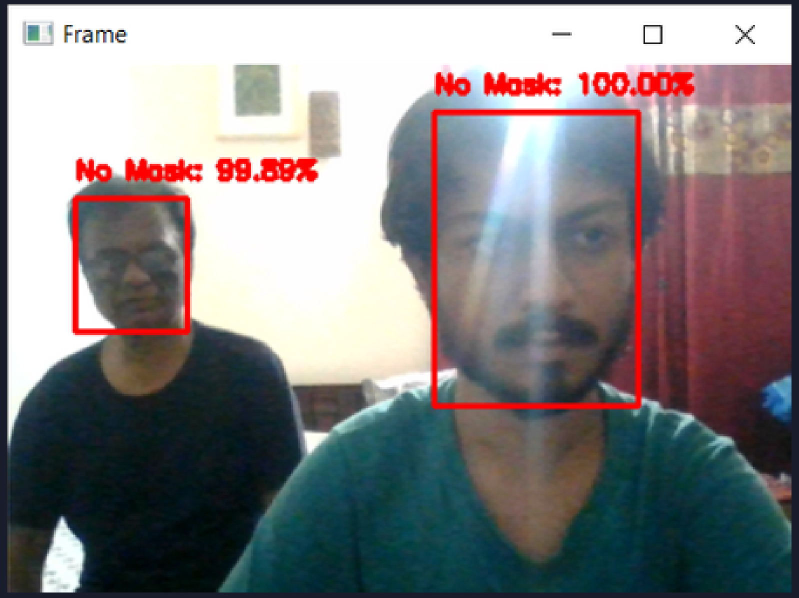
Then first of all we run the code train_mask_detector.py file to verify the training model. Then if it doesn't show any error, then it will create a file named “mask_detector.model”.

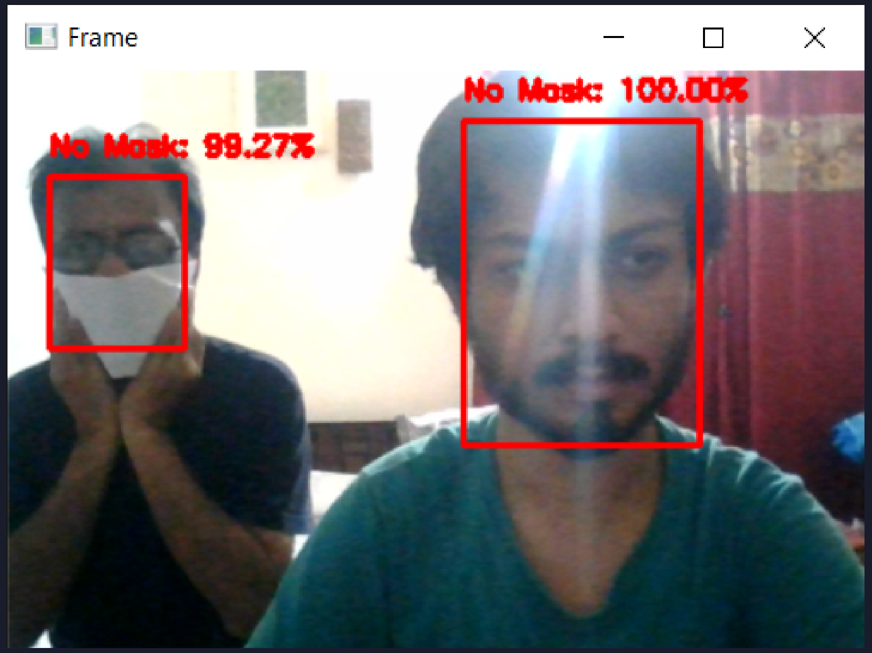
So,we must put all these files in the same
folder or directory.Then we run the administrator command to run the code. Then we must write “python train_mask_detector.py” then after some times, there have been a training and testing for all the images in the folder. Then a file created named “mask_detector.model”. Then our works are almost finished. Then we come to the file named “detect_mask_video.py”. We must run the code in same procedure.
Then a video streaming will show on and the rectangular box will show us the identifying the Mask on or not.

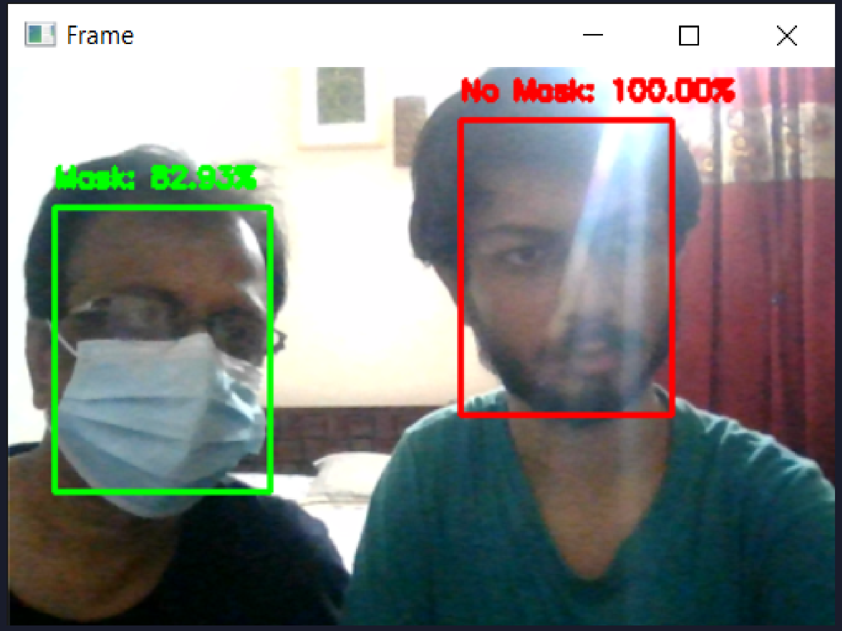
User Interface

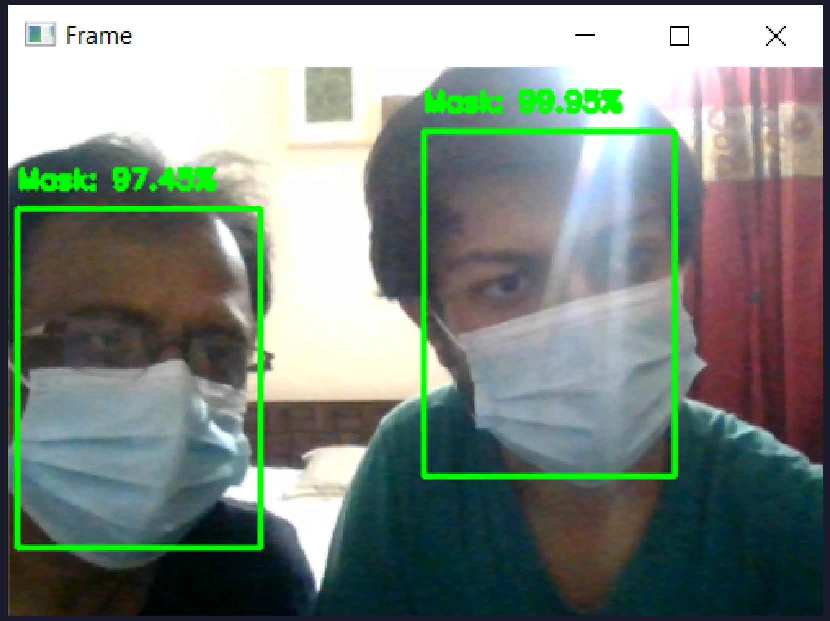




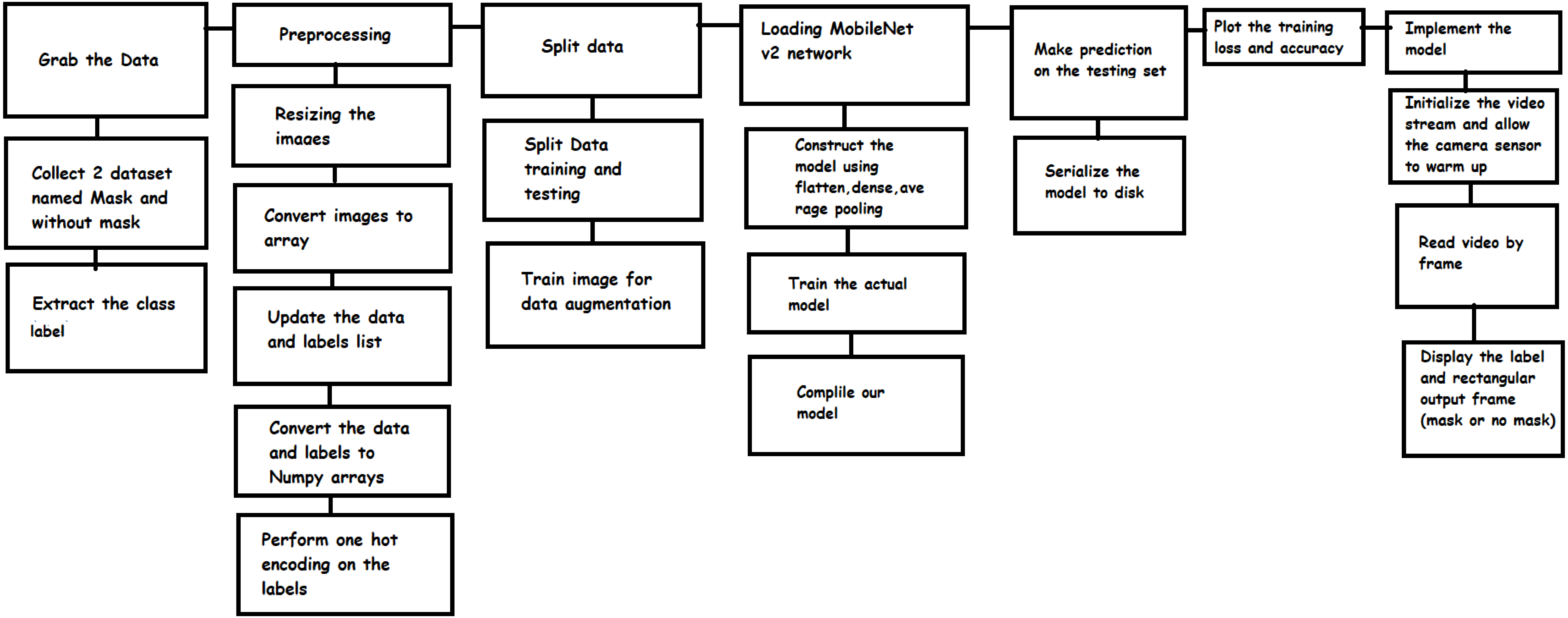








The Main Architecture



- Grab the Data
- Collect 2 dataset named Mask and without mask
- Extract the class label

- Preprocessing
- Resizing the images
- Convert images to array
- Update the data and labels list
- Convert the data and labels to Numpy arrays
- Perform one hot encoding on the labels

- Split data
- Split Data training and testing
- Train image for data augmentation

- Loading MobileNet v2 network
- Construct the model using flatten,dense,ave rage pooling
- Train the actual model
- Complile our model

- Make prediction on the testing set
- Serialize the model to disk

- Plot the training loss and accuracy

- Implement the model
- Initialize the video stream and allow the camera sensor to warm up
- Read video by frame
- Display the label and rectangular output frame (mask or no mask)

Software Needed

1. OpenCv
2. Cuda
3. Keras
4. MobileNet v2
5. Tensorflow
6. Numpy
7. Matplotlib

Language Used

Python is supremely efficient. It allows developers to complete more work using fewer lines of code. The Python code is also easily understandable by humans, which makes it ideal for making Machine Learning models.So we use python to create our model.

Features and Functions

1. AveragePooling 2D: Calculate the average value of a patch of a feature map.It fetches the stride of any matrix and lessen it.
2. Flatten:The 2 dimensional matrix can be converted in to 1 dimentional or vector with this function.Then it connect with the fully-connected layer.
3. Dense: It is a neural network connected deeply,which means each neuron in the dense layer receives input from all neurons of its previous layer.
4. Adam: It is a learning rate optimization algorithm. it can find individual rates for each parameter.Its methods are in 2 parts. One is AdaGrad and other one is RMS propagation.
5. Image_data_generator:It can generate the images in real-time while our training has been proceed.

6.Epoch:Number of batchs to be selected for one epoch

7.Preprocess_input: this input takes the array encoding a batch of images

8.Image_to_array: Here, these images can be converted in to array.

9.Train_test_split: Selection for splitting data arrays into two subsets training data and testing data. if the test_size=0.2 which means that about 80% data will be split for training and rest 20 will be for testing. Besides, random state means the internal random number which decide the training and testing.Its value is maximum time is fixed.

10.os.path.join: it means that the creation of path string by combining the file and directory names.

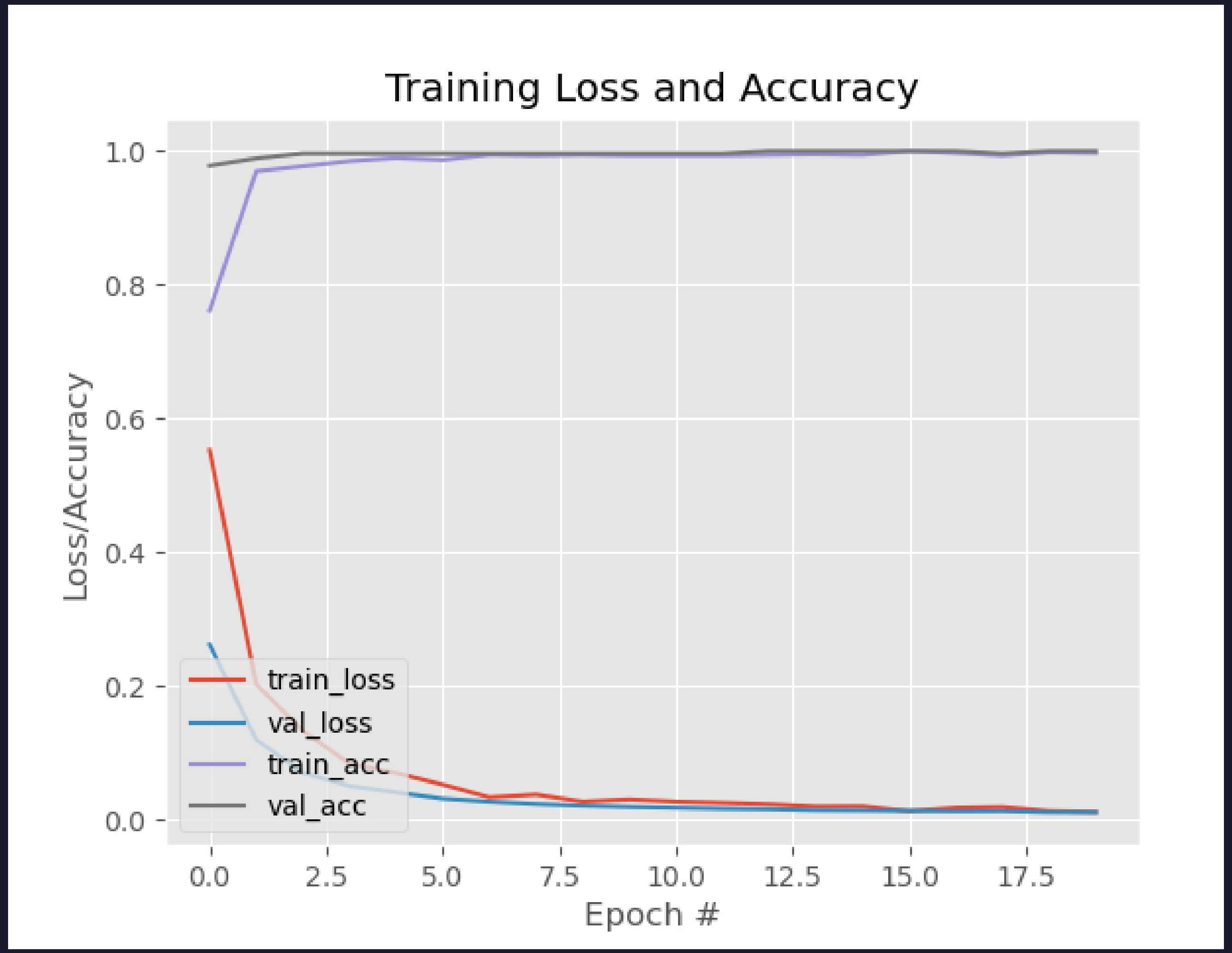
11.fit: its is to train our model with parameter where trainX, target data,validation data, number of epochs.

12.prediction: it will show the actual prediction by viewing array on the probability.

Dataset and performance Analysis

In the system, we fetch the data from kaggle and name the two corresponding folders “dataset”. In the folder, there are the images for training and testing which are “with mask” and “without mask”.In the “with mask” folder there are about 690 images for the model training and on the other hand, about 686 images for “without mask”.In our system, we use the train_test_split to split those images where about 80% for the training and 20% for the testing.

In the UI, we can see that the expected results have been found. There we use the multiple person the detects the mask. It succeed. Besides on the plt.savefig(args["plot"]) where it saved the figure the plot which describe the training loss and accuracy has been shown.It has 99% success for the model. Furthermore, we use the dropout function about 0.5 which means the data has been dropped out to avoid the overfitting graph.So, now we may say that it is a balanced layer.



Challenges

1.Installing Cuda, tensorflow and pip

2.Need high GPU to train model

3.Installing OpenCV in command prompt

4. Installing anaconda

5. Fetching mask on and off data.

6. CPU over heating

Future Works

1.These project can be used to identifying the citizen across the country to detect the wearning mask or not
2.Using OpenCv,and MobileNet v2, we also can indentify the person's office or shopping mall to get permission without wearing mask.
3.This project can be developed to identify how long mask are used in mouth.Is it useable or not.
4.Besides, it can be identify the depth of mask which can tell us if it is protected from corona virus or not.

Discussion

Due to the urgency of controlling COVID-19, the application value and importance of real-time mask detection are increasing. To address this issue, we built these system with quality masked face images and developed by MobileNet v2, a fast and accurate mask detector.The rapid outbreak of COVID-19 has caused serious harm and infected tens of millions of people worldwide. Since there is no specific treatment, wearing masks has become an effective method to prevent the transmission of COVID-19 and is required in most public areas, which has also led to a growing demand for automatic real-time mask detection services to replace manual reminding.In this project,we have included about 1376 images of the person to train our model. Here, about 690 images with mask and 686 images without mask.By the CNN, 4 steps have been utilize though these images and MobileNet v2, and OpenCV.Furthermore, we have train the model in to these images and see the expected results with training loss and accuracy with the help of savefig function on “plot”.Besides, here the dept wise and pointwise operation has been done through these images in MobileNet v2.

Conclusion

In the conclusion, this project shows the identification of mask on and off with accurate predictions.After the training, validation, and testing phase, the model can provide the percentage of people using face mask in some cities with high accuracy.By this model, we can use them to cctv to locate the person and their identification whether their mask on or off.Usage of CNN,MobileNet we easily operate those function and thus we will find a good results.This study is highly crucial for implementation on the Deep learning model and CNN.In future, we will be highly benefited by implementing these models.

Thank you so much for your time and consideration

End